

VB.NET - SEND EMAIL

http://www.tutorialspoint.com/vb.net/vb.net_send_email.htm

Copyright © tutorialspoint.com

VB.Net allows sending E-mails from your application. The **System.Net.Mail** namespace contains classes used for sending emails to a Simple Mail Transfer Protocol (SMTP) server for delivery.

The following table lists some of these commonly used classes:

S.N	Class	Description
1	Attachment	Represents an attachment to an e-mail.
2	AttachmentCollection	Stores attachments to be sent as part of an e-mail message.
3	MailAddress	Represents the address of an electronic mail sender or recipient.
4	MailAddressCollection	Store e-mail addresses that are associated with an e-mail message.
5	MailMessage	Represents an e-mail message that can be sent using the SmtpClient class.
6	SmtpClient	Allows applications to send e-mail by using the Simple Mail Transfer Protocol (SMTP).
7	SmtpException	Represents the exception that is thrown when the SmtpClient is not able to complete a Send or SendAsync operation.

The SmtpClient Class

The SmtpClient class allows applications to send e-mail by using the Simple Mail Transfer Protocol (SMTP).

Following are some commonly used properties of the SmtpClient class:

S.N	Property	Description
1	ClientCertificates	Specify which certificates should be used to establish the Secure Sockets Layer (SSL) connection.
2	Credentials	Gets or sets the credentials used to authenticate the sender.
3	EnableSsl	Specify whether the SmtpClient uses Secure Sockets Layer (SSL) to encrypt the connection.
4	Host	Gets or sets the name or IP address of the host used for SMTP transactions..
5	Port	Gets or sets the port used for SMTP transactions.
6	Timeout	Gets or sets a value that specifies the amount of time after which a synchronous Send call times out.
7	UseDefaultCredentials	Gets or sets a Boolean value that controls whether the DefaultCredentials are sent with requests.

Following are some commonly used methods of the Smtplib class:

S.N	Method & Description
1	Dispose Sends a QUIT message to the SMTP server, gracefully ends the TCP connection, and releases all resources used by the current instance of the Smtplib class.
2	Dispose(Boolean) Sends a QUIT message to the SMTP server, gracefully ends the TCP connection, releases all resources used by the current instance of the Smtplib class, and optionally disposes of the managed resources.
3	OnSendCompleted Raises the SendCompleted event.
4	Send(MailMessage) Sends the specified message to an SMTP server for delivery.
5	Send(String, String, String, String) Sends the specified e-mail message to an SMTP server for delivery. The message sender, recipients, subject, and message body are specified using String objects.
6	SendAsync(MailMessage, Object) Sends the specified e-mail message to an SMTP server for delivery. This method does not block the calling thread and allows the caller to pass an object to the method that is invoked when the operation completes.
7	SendAsync(String, String, String, String, Object) Sends an e-mail message to an SMTP server for delivery. The message sender, recipients, subject, and message body are specified using String objects. This method does not block the calling thread and allows the caller to pass an object to the method that is invoked when the operation completes.
8	SendAsyncCancel Cancels an asynchronous operation to send an e-mail message.
9	SendMailAsync(MailMessage) Sends the specified message to an SMTP server for delivery as an asynchronous operation.
10	SendMailAsync(String, String, String, String) Sends the specified message to an SMTP server for delivery as an asynchronous operation. . The message sender, recipients, subject, and message body are specified using String objects.
11	ToString Returns a string that represents the current object.

The following example demonstrates how to send mail using the Smtplib class. Following points are to be noted in this respect:

- You must specify the SMTP host server that you use to send e-mail. The **Host** and **Port** properties will be different for different host server. We will be using gmail server.
- You need to give the **Credentials** for authentication, if required by the SMTP server.

- You should also provide the email address of the sender and the e-mail address or addresses of the recipients using the **MailMessage.From** and **MailMessage.To** properties respectively.
- You should also specify the message content using the **MailMessage.Body** property.

Example

In this example, let us create a simple application that would send an email. Take the following steps:

1. Add three labels, three text boxes and a button control in the form.
2. Change the text properties of the labels to - 'From', 'To:' and 'Message:' respectively.
3. Change the name properties of the texts to txtFrom, txtTo and txtMessage respectively.
4. Change the text property of the button control to 'Send'
5. Add the following code in the code editor.

```
Imports System.Net.Mail
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' Set the caption bar text of the form.
        Me.Text = "tutorialspoint.com"
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Try
            Dim Smtplib_Server As New Smtplib_Client
            Dim e_mail As New MailMessage()
            Smtplib_Server.UseDefaultCredentials = False
            Smtplib_Server.Credentials = New Net.NetworkCredential("username@gmail.com", "password")
            Smtplib_Server.Port = 587
            Smtplib_Server.EnableSsl = True
            Smtplib_Server.Host = "smtp.gmail.com"

            e_mail = New MailMessage()
            e_mail.From = New MailAddress(txtFrom.Text)
            e_mail.To.Add(txtTo.Text)
            e_mail.Subject = "Email Sending"
            e_mail.IsBodyHtml = False
            e_mail.Body = txtMessage.Text
            Smtplib_Server.Send(e_mail)
            MsgBox("Mail Sent")

            Catch error_t As Exception
                MsgBox(error_t.ToString)
            End Try
        End Sub
    End Sub
```

You must provide your gmail address and real password for credentials.

When the above code is executed and run using **Start** button available at the Microsoft Visual Studio tool bar, it will show following window which you will use to send your emails, try it yourself.

tutorialspoint.com

From:

To:

Message: